

# Standard QA2 procedure @(EU)ARC:

## Current status & future development

---

**Miro Barta,**

in collaboration with

**Wenjuan Liu, Galina Motorina, Ivica Skokic, Roman Brajsa  
& the EU ARC/Czech node Team**

[barta@asu.cas.cz](mailto:barta@asu.cas.cz)

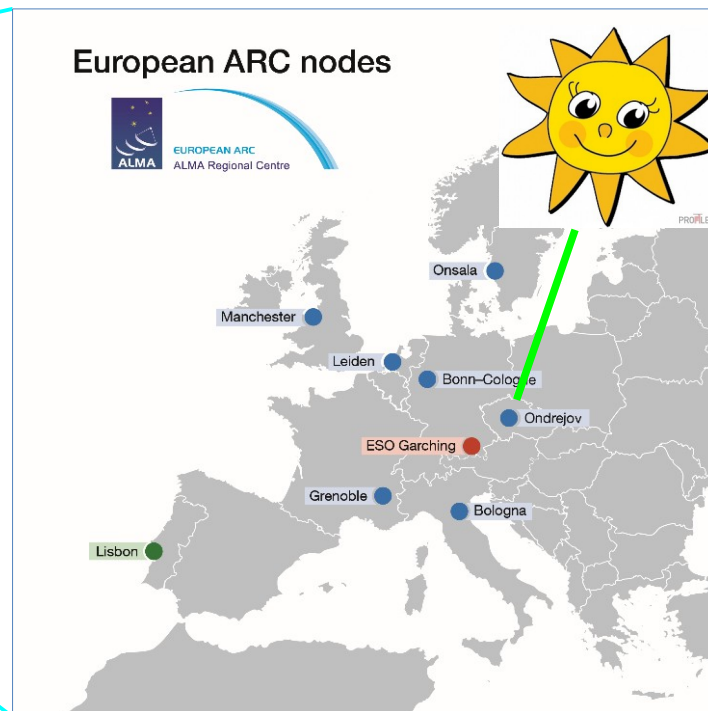
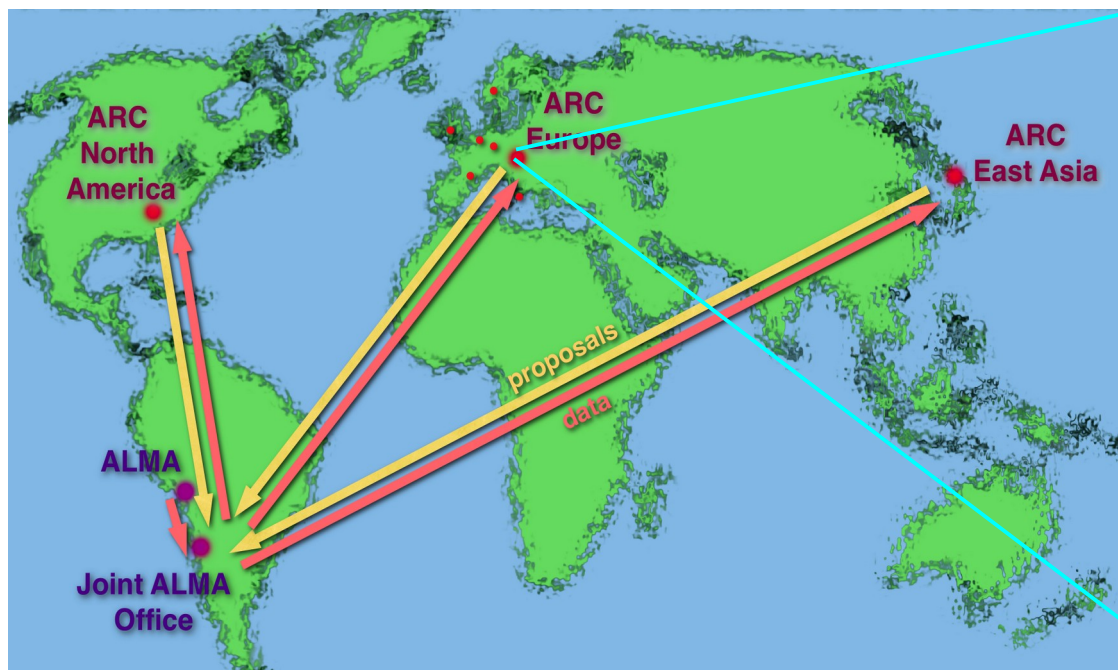


**EUROPEAN ARC**  
ALMA Regional Centre || Czech



**Astronomical  
Institute**  
of the Czech Academy  
of Sciences

- **Standard QA2-passed ALMA products not sufficient as science-ready data**
  - INT and TP data provided separately, no combination/feathering
  - Time-integrated representative image – smeared-out by internal solar dynamics
  - No self-calibration
    - Community effort has started in the direction of calibrated data post-processing – this is why we are here (in Europe: UiO Oslo, Uni Ioannina, Uni Zagreb).
- **Standard QA2 procedure for solar data is fully manual so far**
  - Significant man-power at ARCs (Ondrejov node @EUARC) required.
  - Human-introduced errors while editing the scripts by hand.
  - Lack of robustness and homogeneity.
  - Solar ObsMode referred (up to Cy7) as “non standard” (bad impression).
    - **Convergence towards standard ALMA pipeline highly desirable – development in this direction carried out at EU ARC/ESO.**



ALMA Regional Centers – ARCs:

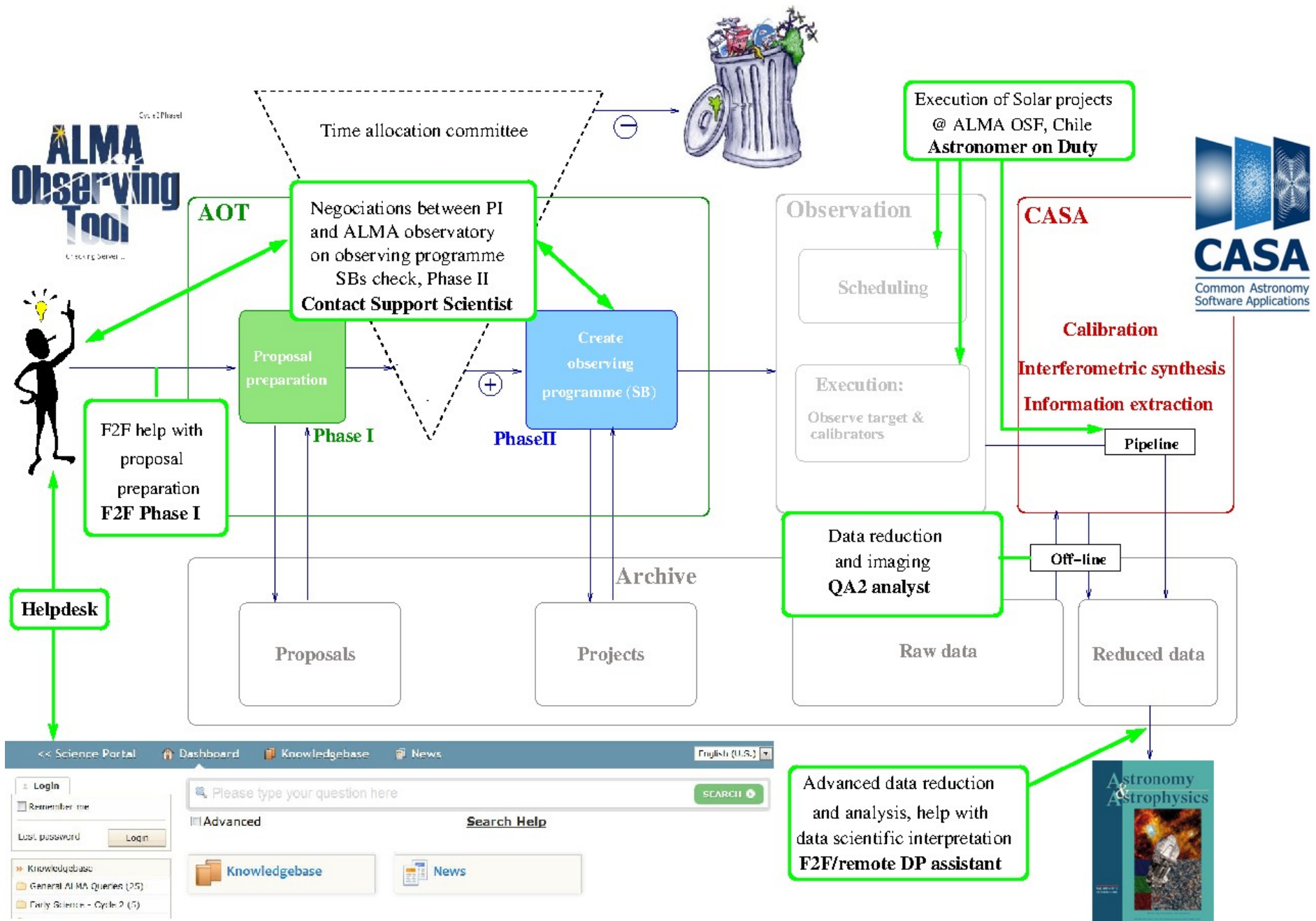
**Supporting infrastructure** – interface between ALMA observatory and user community

Structure of the European ARC:

- Head in ESO Garching
- Seven nodes across Europe
  - ▶ **One in Ondřejov/Prague, Czechia (since 2009)**



EUROPEAN ARC  
ALMA Regional Centre



# Standard solar QA2: Unpopular but necessary “black job”

- **Standard QA2 includes**

- Flagging & calibration of solar data: Results of the manpower and many (sometimes painful) decisions are hidden in the script for calibration).
- Imaging (time integrated :- ( ) & check of the image quality.
- Running **QA2** post-processing **product-evaluation** scripts.

- **Results of the QA2 procedure**

- Products (FITS files)
  - The image is integrated over EB duration (~1h) and takes advantage of multi-frequency synthesis (MFS). Good **starting model** for selfCal.
- Scripts for calibration and imaging (can be invoked from the *scriptForPI*).
  - Script for calibration: One can restore the calibrated measurement set – this is always **the first step for advanced imaging** (time-domain, selfCal).

# Standard solar QA2: Unpopular but necessary “black job”

- **The QA2 should be**
  - Error-free – not only serious mistakes but also violation of the *best practice* procedures should be avoided.
  - Robust – work ideally for all datasets (no need to data-tailored procedure).
  - Homogeneous – all PIs should get the qualitatively the same results.
  - Automated as much as possible
    - **Approaching the non-solar data treatment**





Jump

Search

DataProcessing

TWiki > DataProcessing Web > QAProceduresForSolarINTData

(02 Sep 2018, MiroslavBarta)

[Raw edit](#) [Edit](#) [Attach](#)

## The QA2 & Packaging of Solar Interferometric data Cycle 5

### Cautions

### Known issues

- [SCOPS-5181](#) - E2E5 solar testing.
- [PRTSPR-32758](#) - Incorrect field of view of the synthesized images from the solar MOSAIC data
- [The bug in the "sun\\_reduction\\_util.py"](#)

### Local Solar Experts:

- JAO: Antonio Hales
- EU: Miroslav Barta
- EA: Masumi Shimojo
- NA: Tim Bastian

Since 2018: CZ node maintains Central documentation to the Science Solar ALMA data reduction at [wikis.alma.cl](http://wikis.alma.cl)

ALMA contacts: Catarina Ubach, Dirk Petry

Hello

[Miroslav Barta](#)

Log Out

Create personal sidebar

DataProcessing Web

Create New Topic

Index

Search

Changes

Notifications

RSS Feed

Statistics

Preferences

Webs

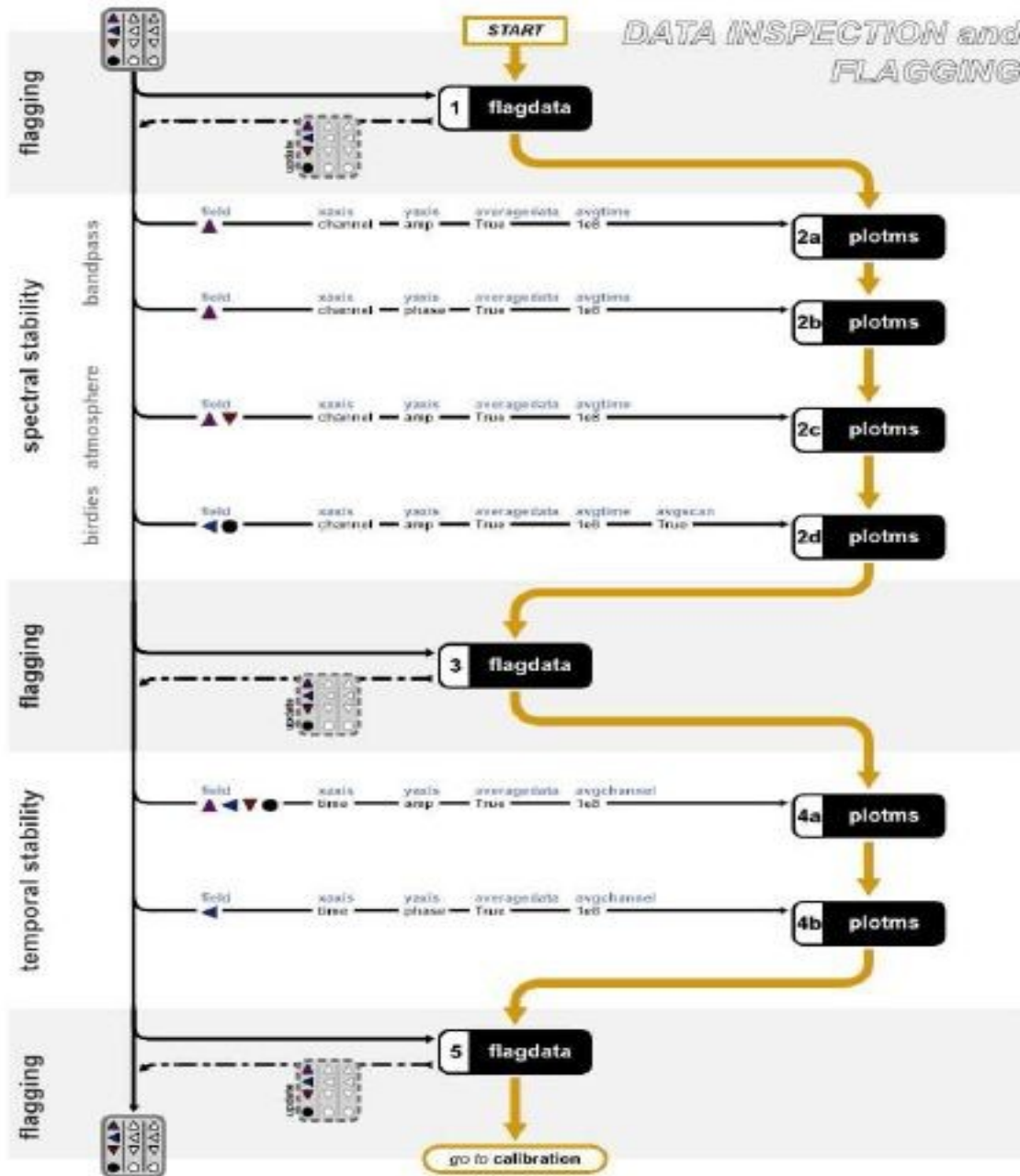
AAER

ADC

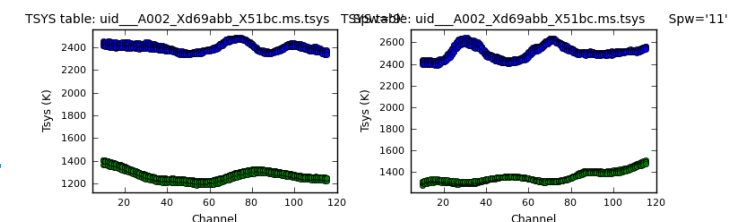
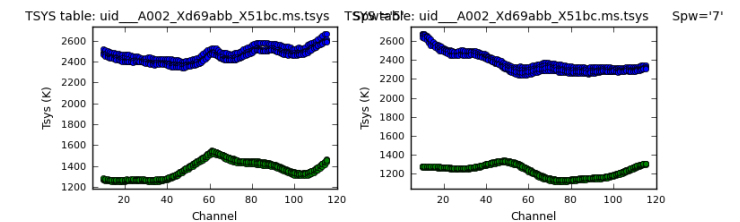
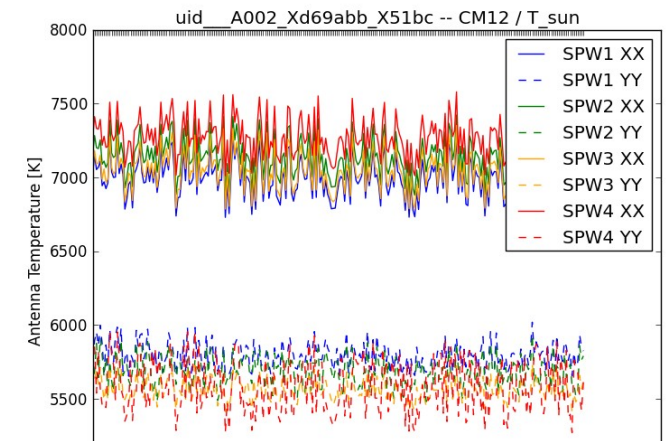
- **“Non-standard” mode**
  - Limited capabilities: Bands 3 & 6 only, TDM only, **configurations limited to  $\leq$  C43-3**, manual cal & imaging,...
  - Combined 12m+7m array, always have complementary TP maps (entire Sun – fast-scanning mode; no regional TP map so far).
  - Using SIS mixer detuning (MD) approach for attenuation of too strong signal from the Sun.
  - Ephemeris targets (solar motion among stars + solar rotation).
  - Literally **manual calibration & imaging** (even no *Script Generator* so far).
- Solar science goals are mostly *Targets of Opportunity* by matter (not technically).
- The Sun is dynamic/highly variable – this breaks down the ‘standard’ interferometry paradigm on signal accumulation.



# Calibration & imaging: Flagging

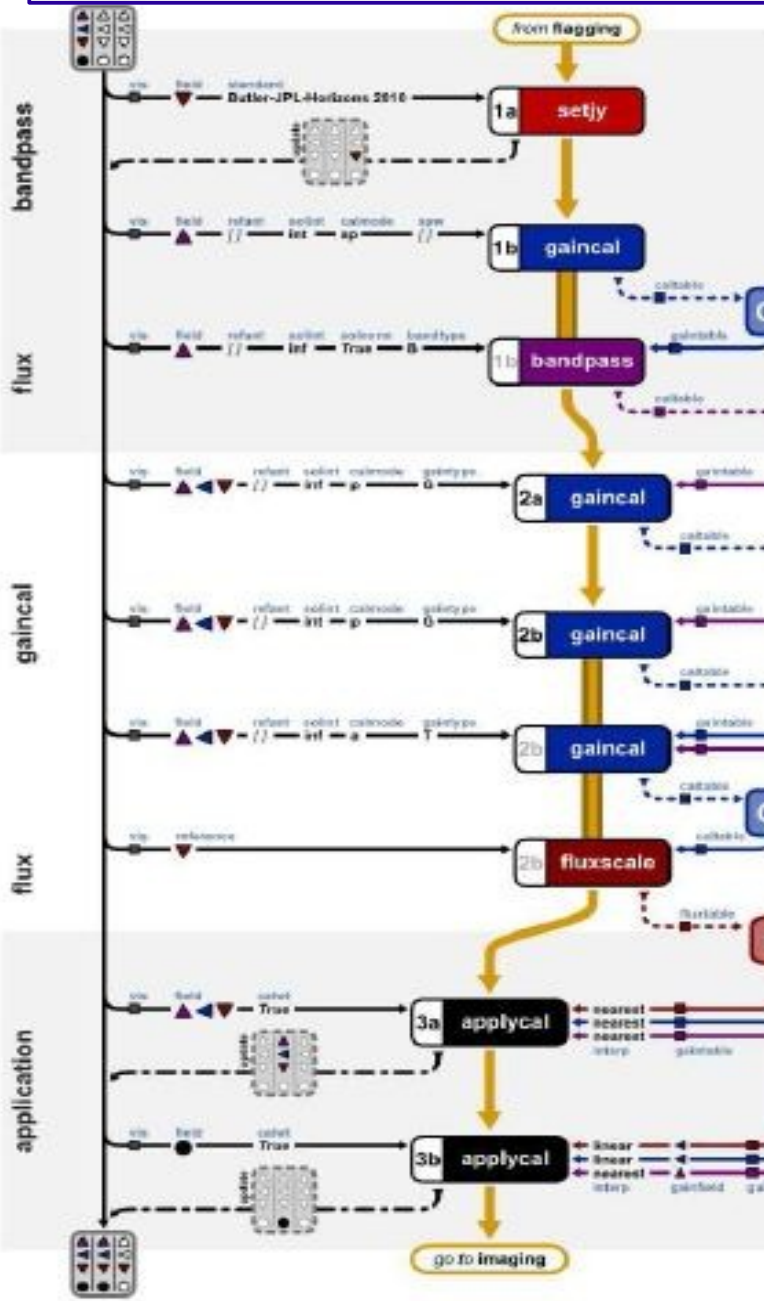


- Solar mode: process demanding on manpower, experience required
- ‘Standard modes’: Heuristics implemented in ALMA pipeline (~75% success rate)
- Our QA2 mission: **Preserve as much as possible data**, remove only the bad ones → many branching points (try and see).
- Opportunity for AI (in far future).





# Calibration & imaging: Editing parameters in scripts



CALIBRATION

The screenshot shows an Emacs editor window titled 'emacs@momentum'. The window displays a list of calibration and observation logs. The logs are organized into several sections, each starting with a time range and a list of parameters. The parameters include 'CALIBRATE\_ATMOSPHERE#AMBIENT', 'CALIBRATE\_ATMOSPHERE#HOT', 'CALIBRATE\_ATMOSPHERE#OFF\_SOURCE', 'CALIBRATE\_WVR#AMBIENT', 'CALIBRATE\_WVR#HOT', 'CALIBRATE\_WVR#OFF\_SOURCE', 'CALIBRATE\_BANDPASS#ON\_SOURCE', 'CALIBRATE\_WVR#ON\_SOURCE', 'CALIBRATE\_POINTING#ON\_SOURCE', 'CALIBRATE\_WVR#ON\_SOURCE', 'CALIBRATE\_PHASE#ON\_SOURCE', 'CALIBRATE\_WVR#ON\_SOURCE', 'OBSERVE\_TARGET#OFF\_SOURCE', and 'OBSERVE\_TARGET#ON\_SOURCE'. The logs also include observation IDs, dates, and times.

```
0.576, 0.576, 0.576, 0.576, 0.576] [CALIBRATE_ATMOSPHERE#AMBIENT,CALIBRATE_ATMOSPHER
E#HOT,CALIBRATE_ATMOSPHERE#OFF_SOURCE,CALIBRATE_WVR#AMBIENT,CALIBRATE_WVR#HOT,CALIBRA
TE_WVR#OFF_SOURCE]
14:09:15.3 - 14:14:18.4 5 1 J1256-0547 6088150 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [CALIBRATE_BANDPASS#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
14:15:20.4 - 14:17:23.3 6 2 J1745-0753 2029400 [4
,25,26,27,28,29,30,31,32,33,34,35,36] [1.15, 0.016, 0.016, 0.016, 0.016, 2.02, 1.01,
2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRATE_POINTING#ON_SOURCE,CALIBRATE_WVR#ON_S
OURCE]
14:18:05.2 - 14:19:06.1 7 3 J1733-1304 1217600 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
14:19:20.1 - 14:19:37.3 8 0 Sun_10 365950 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 0.576, 0.576, 0.576,
0.576, 0.576, 0.576, 0.576, 0.576] [CALIBRATE_ATMOSPHERE#AMBIENT,CALIBRATE_ATMOSPHER
E#HOT,CALIBRATE_ATMOSPHERE#OFF_SOURCE,CALIBRATE_WVR#AMBIENT,CALIBRATE_WVR#HOT,CALIBRA
TE_WVR#OFF_SOURCE]
14:20:03.8 - 14:27:18.9 9 0 Sun_10 487000 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#OFF_SOURCE]
14:20:03.8 - 14:27:18.9 9 4 Sun_10 121750 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#ON_SOURCE]
14:20:03.8 - 14:27:18.9 9 5 Sun_10 121750 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#ON_SOURCE]
14:20:03.8 - 14:27:18.9 9 6 Sun_10 121750 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#ON_SOURCE]
14:20:03.8 - 14:27:18.9 9 7 Sun_10 121750 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#ON_SOURCE]
14:20:03.8 - 14:27:18.9 9 8 Sun_10 121750 [0
,1,2,3,4,5,6,7,8,9,10,11,12] [0.016, 0.016, 0.016, 0.016, 1.15, 2.02, 1.01, 2.02, 1.
01, 2.02, 1.01, 2.02, 1.01] [OBSERVE_TARGET#ON_SOURCE]
-:--- uid A002 Xd69abb X51bc listobs.txt 2% (14,207) (Text Fill)
```

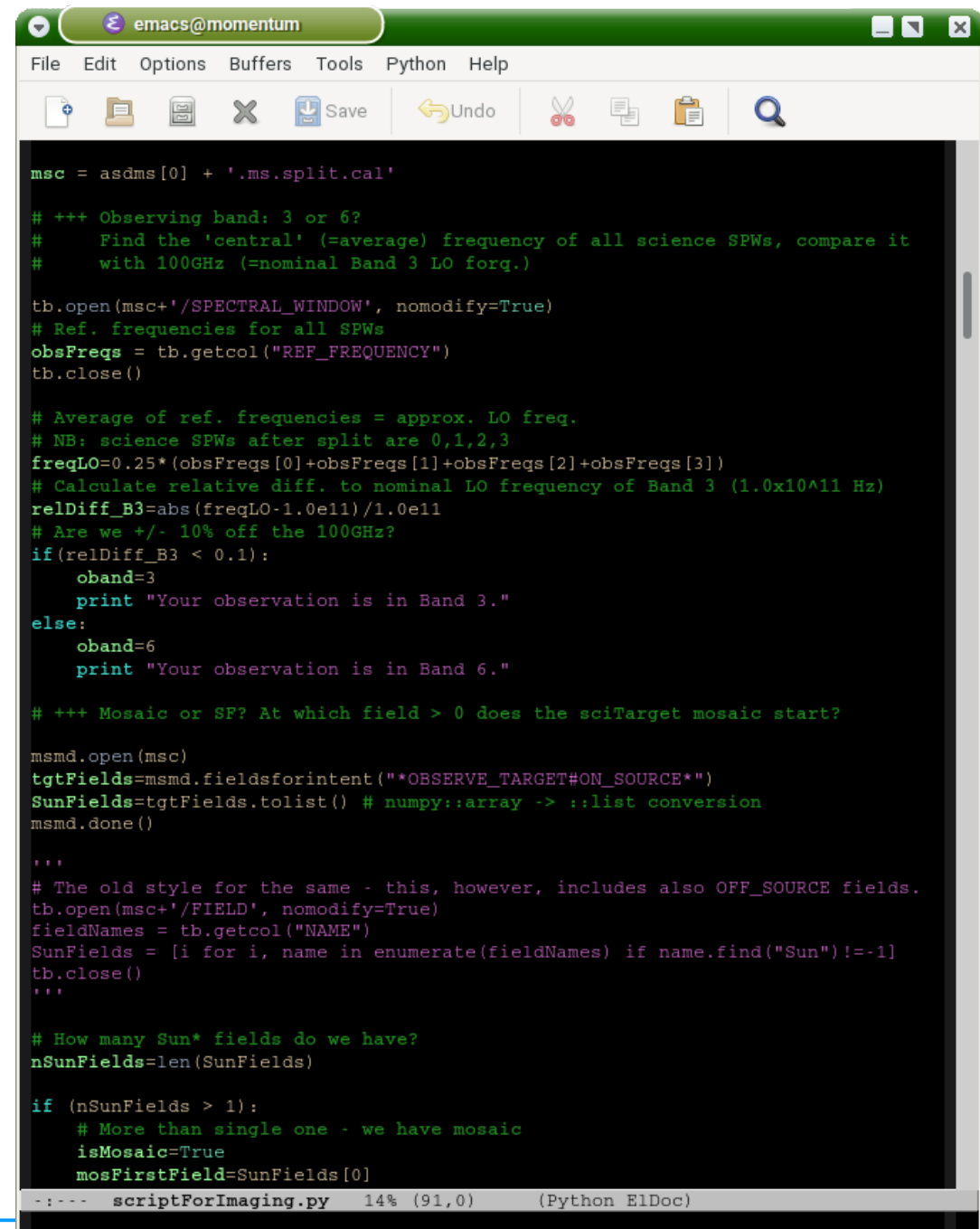


## Manual calibration and imaging cook book (up to Cy 5):

- 1) Take the sample script working well for some EB as a template
- 2) Do manual diagnostics (listobs(), browsetable(),...) of the MS and set/edit manually the parameters in the cal and img scripts.
  - Many of them: Band 3/6, refAnt, refTime, cal and tgt fields & scans in gaincal()/applycal(), atm lines flagging (different for different bands), mosaic/single-field, img size and cell, gridder mode,...
  - Great potential for mistake/omission
  - Demanding on manpower
  - Dependent on QA2 analyst (calibrated data are not homogeneous across the ARCs).
- 3) Do manual analysis/selection and flagging
- 4) Run the cal script step-wise (in 5 batches), modify the params before next step.
- 5) Perform QA2 analysis/report
- 6) Run imaging script (after some parameter modifications based on QA2 report).
- 7) Report on AQUA, prepare package for final check (at ESO) and delivery.

## “Universal scripts” since Cy 5

- Auto-detection blocks inside the scripts – automated set of all parameters (antPos correction table has to be copy/paste inserted).
- ✓ No need for manual input from the QA2 analyst → reduced manpower, omissions; homogeneity of calibration
- ✗ Big parts of identical code to be copied into all scripts – loss of clarity.
- ➔ What to do: Remove the auto-detect parts into separate code, let this program to write the cal & img scripts for us.



```
emacs@momentum
File Edit Options Buffers Tools Python Help
+ Save Undo
msc = asdms[0] + '.ms.split.cal'

# +++ Observing band: 3 or 6?
# Find the 'central' (=average) frequency of all science SPWs, compare it
# with 100GHz (=nominal Band 3 LO freq.)

tb.open(msc+'/SPECTRAL_WINDOW', nomodify=True)
# Ref. frequencies for all SPWs
obsFreqs = tb.getcol("REF_FREQUENCY")
tb.close()

# Average of ref. frequencies = approx. LO freq.
# NB: science SPWs after split are 0,1,2,3
freqLO=0.25*(obsFreqs[0]+obsFreqs[1]+obsFreqs[2]+obsFreqs[3])
# Calculate relative diff. to nominal LO frequency of Band 3 (1.0x10^11 Hz)
relDiff_B3=abs(freqLO-1.0e11)/1.0e11
# Are we +/- 10% off the 100GHz?
if(relDiff_B3 < 0.1):
    oband=3
    print "Your observation is in Band 3."
else:
    oband=6
    print "Your observation is in Band 6."

# +++ Mosaic or SP? At which field > 0 does the sciTarget mosaic start?

msmd.open(msc)
tgtFields=msmd.fieldsforint("OBSERVE_TARGET#ON_SOURCE")
SunFields=tgtFields.tolist() # numpy::array -> ::list conversion
msmd.done()

'''
# The old style for the same - this, however, includes also OFF_SOURCE fields.
tb.open(msc+'/FIELD', nomodify=True)
fieldNames = tb.getcol("NAME")
SunFields = [i for i, name in enumerate(fieldNames) if name.find("Sun")!=-1]
tb.close()
'''

# How many Sun* fields do we have?
nSunFields=len(SunFields)

if (nSunFields > 1):
    # More than single one - we have mosaic
    isMosaic=True
    mosFirstField=SunFields[0]

-:--- scriptForImaging.py 14% (91,0) (Python ElDoc)
```

## Manual calibration and imaging cook book since Cy 5:

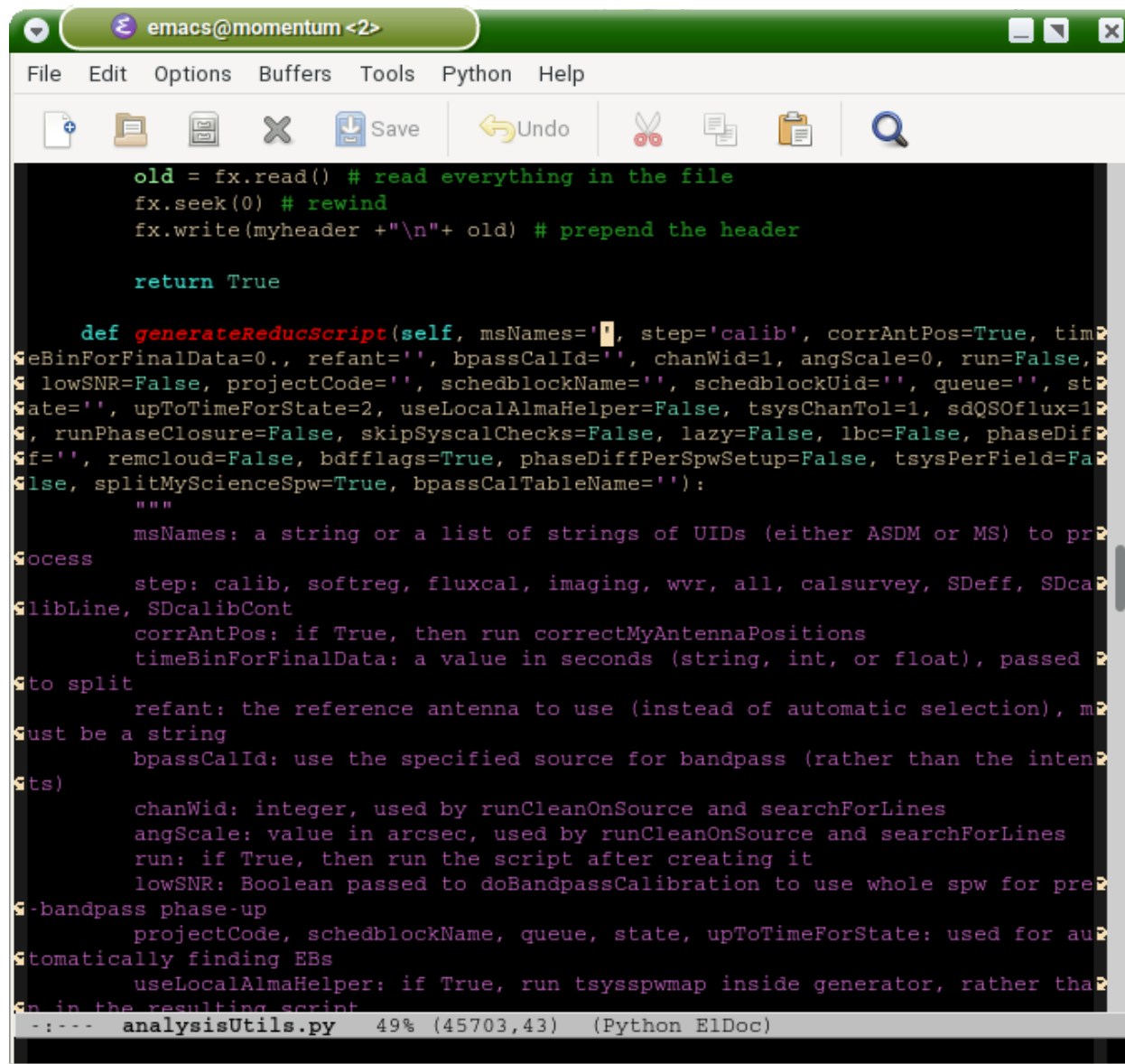
- 1) Copy the “universal scripts” from the AIV mirror to your QA2 workdir.
- 2) Do manual analysis/selection and flagging
- 3) Run 1<sup>st</sup> half of the cal script, copy/paste its outcome in console to the 2<sup>nd</sup> half and run the 2<sup>nd</sup> half
- 4) Perform QA2 analysis/report
- 5) Run imaging script.
- 6) Report on AQUA, prepare package for final check (at ESO) and delivery.



- Cycles 0-2: Fully manual cal & img, editing the scripts (in solar mode: till Cy5)
- Cycle 3, partly 4: **Script Generator**
- Since Cy 4: (almost all dataset) ALMA CASA pipeline calibration and imaging (pipelined imaging being started a bit later).

## Script Generator (non-solar)

- Main developer and maintainer Eric Villard, since 2020 Dirk Petry @ESO
- “Father of all scripts”
- Diagnoses the MS and writes the tailored cal & img scripts (img script generator contributed by A. Borkar from our ARC node)
- Part of the *analysisUtils.py* package.



```
emacs@momentum <2>
File Edit Options Buffers Tools Python Help
+ Save Undo
old = fx.read() # read everything in the file
fx.seek(0) # rewind
fx.write(myheader + "\n" + old) # prepend the header

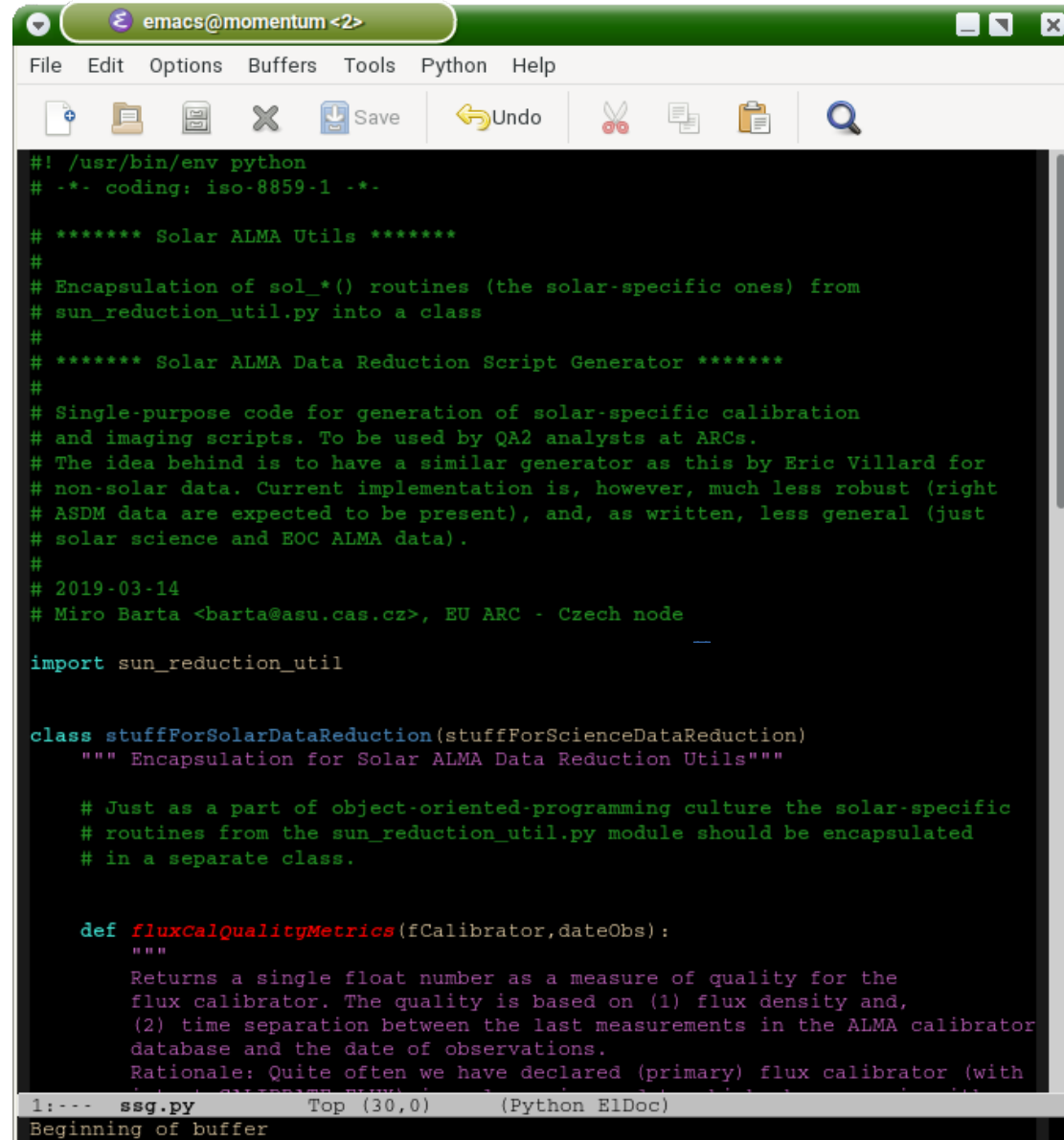
return True

def generateReducScript(self, msNames='', step='calib', corrAntPos=True, timeBinForFinalData=0., refant='', bpassCalId='', chanWid=1, angScale=0, run=False, lowSNR=False, projectCode='', schedblockName='', schedblockUid='', queue='', state='', upToTimeForState=2, useLocalAlmaHelper=False, tsysChanTol=1, sdQSOflux=1, runPhaseClosure=False, skipSyscalChecks=False, lazy=False, lbc=False, phaseDiff='', remcloud=False, bdfFlags=True, phaseDiffPerSpwSetup=False, tsysPerField=False, splitMyScienceSpw=True, bpassCalTableName=''):
    """
    msNames: a string or a list of strings of UIDs (either ASDM or MS) to process
    step: calib, softreg, fluxcal, imaging, wvr, all, calsurvey, SDeff, SDcalibLine, SDcalibCont
    corrAntPos: if True, then run correctMyAntennaPositions
    timeBinForFinalData: a value in seconds (string, int, or float), passed to split
    refant: the reference antenna to use (instead of automatic selection), must be a string
    bpassCalId: use the specified source for bandpass (rather than the intended)
    chanWid: integer, used by runCleanOnSource and searchForLines
    angScale: value in arcsec, used by runCleanOnSource and searchForLines
    run: if True, then run the script after creating it
    lowSNR: Boolean passed to doBandpassCalibration to use whole spw for pre-bandpass phase-up
    projectCode, schedblockName, queue, state, upToTimeForState: used for automatically finding EBs
    useLocalAlmaHelper: if True, run tsysspwmap inside generator, rather than in the resulting script
    """
    analysisUtils.py 49% (45703,43) (Python ElDoc)
```

- Based on Python class derived/sub-classed from the E.V.s' *stuffForScienceDataRedution*
- Implements solar-specific calibration steps.
- The same usage as standard SG

## Some solar specifics

- $T_{\text{ant}}+T_{\text{sys}}$  calibration – `sol_ampcal2()` CPU-time demanding (~2days), **we work on MPI paralelization.**
- Flux calibrator: The nominal FluxCal is frequently of low quality (old measurement or weak) – better to use B-cal for amplitude calibration. Implement **quality metrics to choose automatically.**



```
emacs@momentum <2>
File Edit Options Buffers Tools Python Help
+ Save Undo
#!/usr/bin/env python
# -*- coding: iso-8859-1 -*-

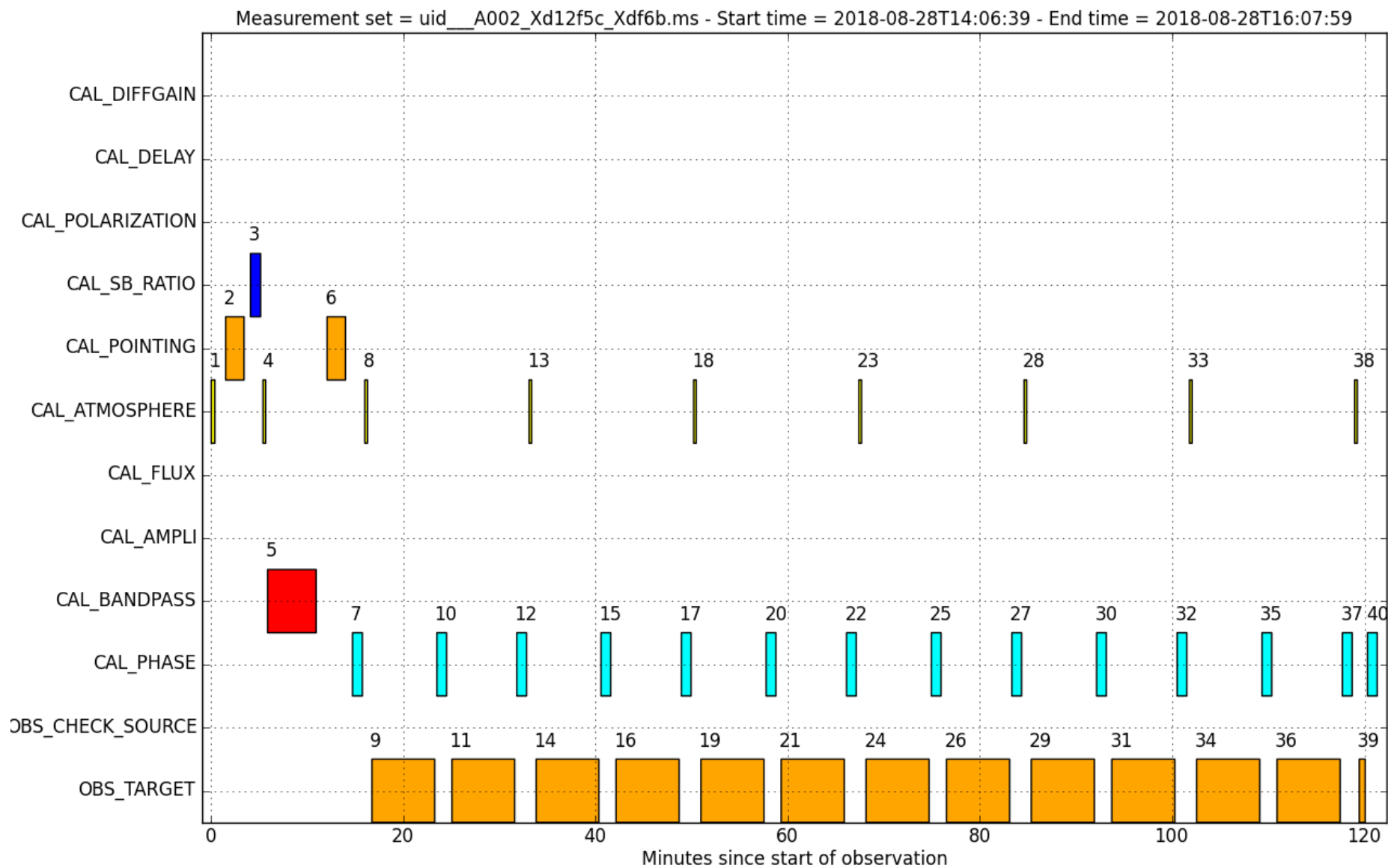
# ***** Solar ALMA Utils *****
#
# Encapsulation of sol_*( ) routines (the solar-specific ones) from
# sun_reduction_util.py into a class
#
# ***** Solar ALMA Data Reduction Script Generator *****
#
# Single-purpose code for generation of solar-specific calibration
# and imaging scripts. To be used by QA2 analysts at ARCs.
# The idea behind is to have a similar generator as this by Eric Villard for
# non-solar data. Current implementation is, however, much less robust (right
# ASDM data are expected to be present), and, as written, less general (just
# solar science and EOC ALMA data).
#
# 2019-03-14
# Miro Barta <barta@asu.cas.cz>, EU ARC - Czech node

import sun_reduction_util

class stuffForSolarDataReduction(stuffForScienceDataReduction)
    """ Encapsulation for Solar ALMA Data Reduction Utils"""

    # Just as a part of object-oriented-programming culture the solar-specific
    # routines from the sun_reduction_util.py module should be encapsulated
    # in a separate class.

    def fluxCalQualityMetrics(fCalibrator,dateObs):
        """
        Returns a single float number as a measure of quality for the
        flux calibrator. The quality is based on (1) flux density and,
        (2) time separation between the last measurements in the ALMA calibrator
        database and the date of observations.
        Rationale: Quite often we have declared (primary) flux calibrator (with
        1:--- ssg.py Top (30,0) (Python ElDoc)
Beginning of buffer
```



- Being a small community and using non-standard mode we are behind the ‘mainstream’ development of ALMA tools (the same situation is, e.g., with the VLBI mode).
- Developed *Solar Script Generator* represents an intermediate step towards future full integration of the solar science data reduction into ALMA pipeline.
- Our mission as the **QA2 analysts / ARC service: Do the “black job”** – provide as much as possible high quality **primary calibrated data** for possible further processing (discard problematic data but flag only what is necessary, try to preserve as much as possible). Calibration procedure should be **robust and homogeneous**, working the same way for **all the solar datasets**. We are attempting to make it as fast as possible (reduce manpower). **We have to guarantee that the primary data are of enough high quality**, fulfilling the science requirements stated in AOT. Outcome: (generally applicable) **Solar Script Generator**, to become **part of standard ALMA pipeline** later.
- (Advanced) User point of view: To have their **individual** dataset **imaged the best way**, mine the maximum from the primary calibrated data. Outcome: (individual) advanced imaging scripts include time-domain imaging, self-calibration and TP/INT data combinaton – **this workshop aims mostly at this goal**.

# Towards hi-resolution solar ALMA images:

Overcoming current *Solar ObsMode* limitations  
(*Development Study 2019* proposal)

---

Miro Barta, Ivica Skokic & Roman Brajsa, EU ARC – Czech node

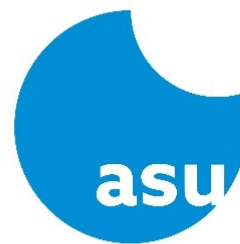
in collaboration with

Tim Bastian & Dale Gary, NRAO / NA ARC, Masumi Shimojo, NAOJ / EA ARC  
& the Solar ObsMode Development Team

[barta@asu.cas.cz](mailto:barta@asu.cas.cz)



EUROPEAN ARC  
ALMA Regional Centre || Czech

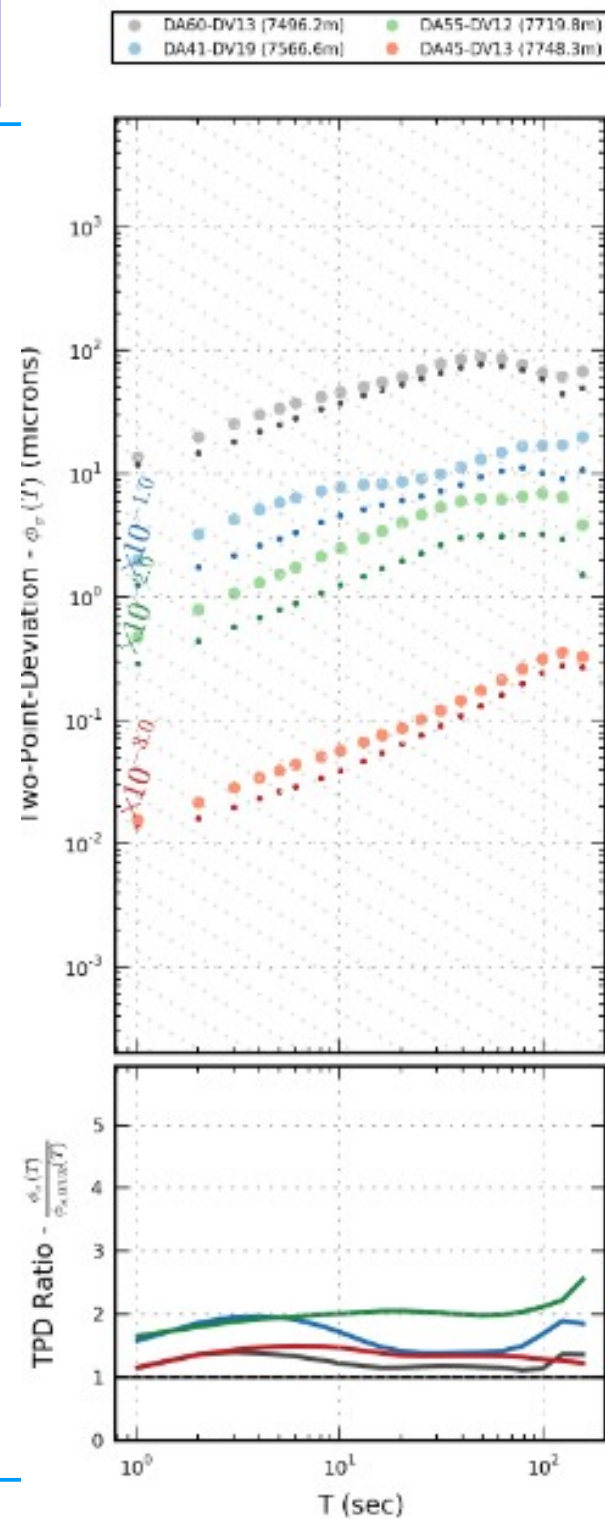
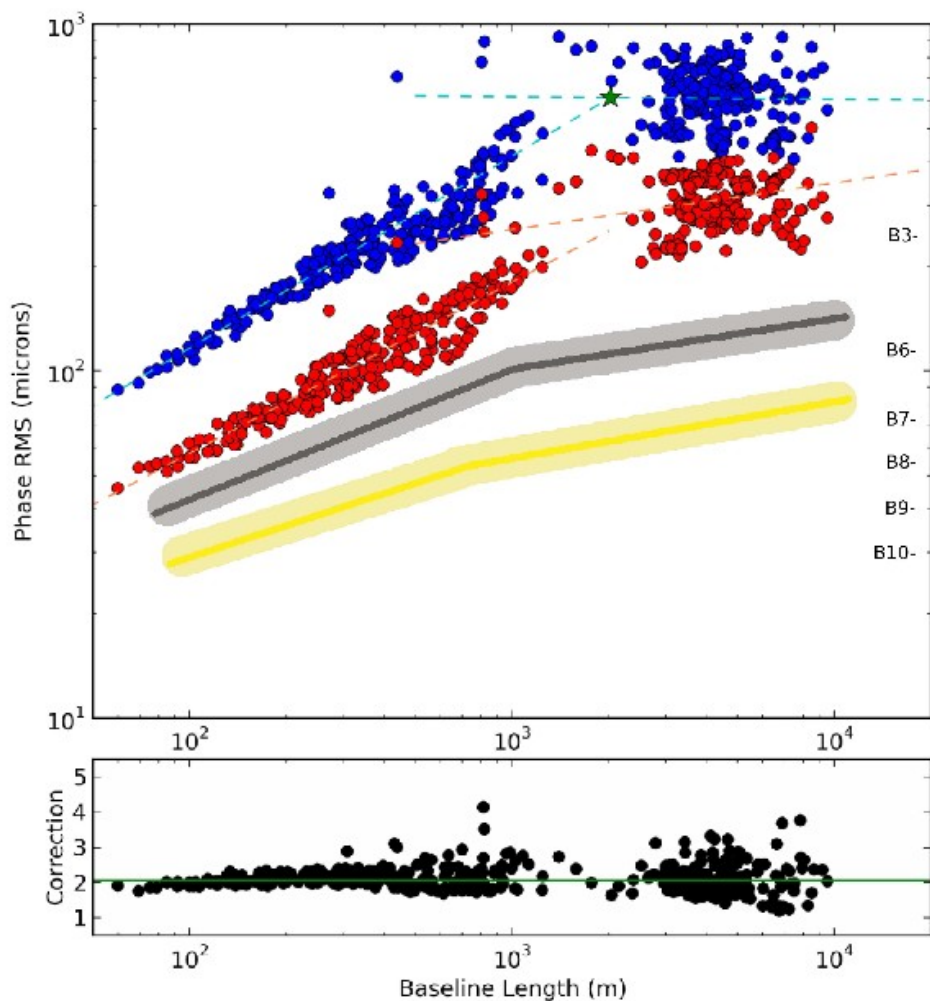


Astronomical  
Institute  
of the Czech Academy  
of Sciences



# Suggested ways out – preliminary ideas

Shorter integrations + selfcal?



L.Maud, R. Tilanus, et al. - ALMA Memo 606 + A&A2017